International Journal of Computer Networks and Distributed Computing
Vol. 1, Issue 1

Digital Image Inpainting Using Cellular Sensor Network and Target Tracking Using RSU Side Units

Usha Kiran[1], Om Prakash Yadav[2]

Assistant Professor, Dept.of CSE, CSIT, Durg, India[1]
Associate Professor[2] , Dept.of CSE, CSIT, Durg, India,[2]

ABSTRACT: Digital Image Inpainting is challenging and interesting research area, because one has to restore the area which is not visible but important to visually complete the image. This technique has found widespread use in applications such as restoration, error recovery, multimedia editing, and video privacy protection. Because of the strong human visual perception, a very effective technique is required for digital image inpainting. Most automatic techniques are computationally intensive and unable to repair large holes. Existing methods use interpolation methods where surrounding information is not adequate for image interpolation and chain codes for contour matching for small damaged area reconstruction. But, reconstructed image has not given up to the mark results. This paper proposed an effective inpainting technique in order to improve the inpainting result. The method proposed in this paper uses Run Length Coding for shape tracking along with CNN approach, because Run Length coding track the shape of an image, which is better than the several methods available for shape tracking.

Keywords: Image inpainting, Cellular Neural Network, Digital images, contour matching, Run Length Code

I. INTRODUCTION

Reconstruction of missing or damaged portions of images is an ancient practice used extensively in artwork restoration. This activity, also known as inpainting or retouching, consists of filling in the missing areas or modifying the damaged ones in a manner non-detectable by an observer not familiar with the original images. The goal of inpainting algorithms varies, depending on the application, from making the inpainted parts look consistent with the rest of the image, to making them as close as possible to the original image, restoration of photographs, films and paintings, to removal of occlusions, such as text, subtitles, stamps and advertisements from images. In addition, inpainting can also be used to produce special effects. While, traditionally skilled artists have performed image inpainting manually, currently digital techniques are used, e.g. for automatic restoration of scratched films.

1.1 DIGITAL IMAGE INPAINTING TECHNIQUES

As a first step the user manually selects the portions of the image that will be restored. Then image restoration is done automatically, by filling these regions in with new information coming from the surrounding or cell in our case. In order to produce a perceptually plausible reconstruction, an inpainting technique must attempt to continue the isophotes (line of equal gray value) as smoothly as possible inside the reconstruction region. In other words the missing region should be inpainted so that inpainted gray value and gradient extrapolate the gray value and gradient outside the region. Several inpainting methods are based on the above ideas.

Bertalmio et al. first introduced the notion of digital image inpainting and used third order partial differential equations (PDE) [4], [5], [9] to diffuse the known image information into the missing regions. Later, this inpainting approach was modified to take into account the direction of the level lines, called isophotes, and to relate it to the Navier-Stokes flow [6], [7, [21].

This operation propagates information into the masked region while preserving the edges. Anisotropic diffusion is used to preserve edges across the inpainted regions [13]. For further discussion of various methods, see the recent survey articles [10], [19]. The algorithms proposed in the literature differ depending on the assumptions made about the properties of the image. For example, the total variation (TV) inpainting model proposed, based on the Euler–Lagrange equation, employs anisotropic diffusion based on the contrast of the isophotes inside the inpainting domain [8]. This model, designed for inpainting small regions, does a good job at removing noise, but does not connect broken edges (single lines embedded in a uniform background). The Curvature-Driven Diffusion (CDD) model, extends the TV algorithm to also take into account geometric information of isophotes when defining the ‘strength’ of the diffusion process, thus allowing the inpainting to proceed over larger areas. Although some of the broken edges are connected by the CDD approach, the resulting criteria for stopping the inpainting, the process is constantly applied to all masked pixels, regardless of the local smoothness of the region. As a result, computationally expensive operations might be unnecessarily performed, resulting in lengthy processing time. Thus, although non-linear PDE-based image restoration methods have the potential of

systematically preserving edges, fast numerical implementations are difficult to design.

II. CONTOUR IDEAS USING RUN LENGTH CODE

Contour tracing is one of many preprocessing techniques performed on digital images in order to extract information about their general shape. Once the contour of a given pattern is extracted, its different characteristics will be examined and used as features which will later on be used in pattern classification. Therefore, correct extraction of the contour will produce more accurate features which will increase the chances of correctly classifying a given pattern. The contour pixels are generally a small subset of the total number of pixels representing a pattern. Therefore, the amount of computation is greatly reduced when we run **feature extracting algorithms** on the contour instead of on the whole pattern. Since the contour shares a lot of features with the original pattern, the feature extraction process becomes much more efficient when performed on the contour rather on the original pattern. In conclusion, **contour tracing** is often a major contributor to the efficiency of the feature extraction process an essential process in the field of pattern recognition. For images retrieval, low-level visual features are color, texture, and shape. Among these features, shape is the most important because it represents significant regions or relevant objects in images. Extensive work has been done in shape retrieval.

In general, shape representations are classified into two categories: boundary-based and region-based. The first one describes the considered region by using its external characteristics (i.e. the pixels along the object boundary) while the second one represents the considered region by using its internal characteristics (i.e. the pixels contained in the region). Several shape description approaches have been developed in the two categories. For example, area, compactness, is bounding box for the region-based category, and perimeter, curvature for the boundary-based category. Several techniques such as chain code, crack code and run length code etc. are existing to represent the region or object by describing its contour.

2.1 RUN LENGTH CODING

Run length coding used in [20] is a simple yet useful technique in lossless coding. Run Length Coding (RLC) is a conceptually simple form of compression. RLC consists of the process of searching for repeated runs of a single symbol in an input stream, and replacing them by a single instance of the symbol and a run count. Run-length coding is performed in order to extract the continuous flat parts from the contour. With run length code and thresholding, we can get the lengths of the flat segments of the contour, which are used for shape comparisons. From the flat segment lengths, we can construct a vector. The length of the vectors may not be equal. To compare vectors with different lengths, we use down sample to make them have the same length. To compare the down-sampled vectors, circular shifts are performed. Finally the minimal standard deviation of the ratio of the shifted vectors is used for the measurement of the shapes.

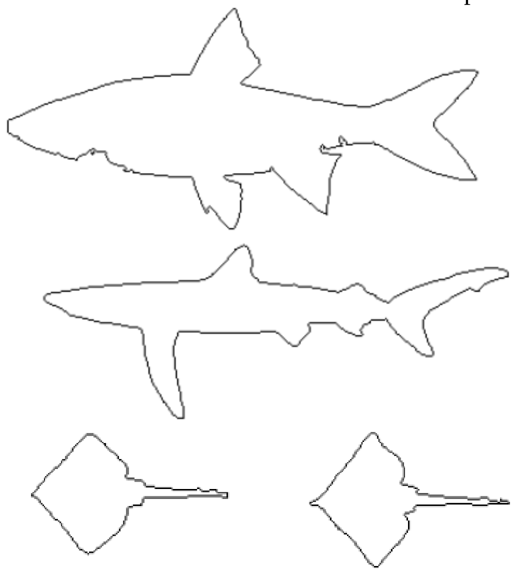

Fig-1 Experimental result of shape tracking

2.2 CELLULAR NEURAL NETWORK

Cellular Neural Networks (CNN) is analog, continuous time, nonlinear dynamic systems and formally belongs to the class of recurrent Neural Networks. Since their introduction in 1988 by Chua and Yang [3], they have been the subjects of intense research. The Cellular Neural Network (CNN) is an artificial neural network of the nearest neighbor interaction type. It has been widely used for image processing, pattern recognition, moving object detection, target classification, signal processing, augmented reality and solving partial differential equations etc. The Cellular Neural Network Complementary Metal Oxide Semiconductor CMOS array was implemented by Anguita. The design of a Cellular Neural Network template is an important problem, and has received wide attention in the recent years. This paper reports an efficient Algorithm exploiting the latency properties of Cellular Neural Networks along with popular numerical approximation algorithms. The dynamic equation of a cell $C(i, j)$ in an M x N Cellular Neural Network is given by Chua and Yang [3].

III. THE RCONSTRUCTION OF DAMAGED IMAGE USING CNN METHOD

P.Elango a and K.Murugesan [2] proposed method using CNN Model, In which this paper is going to perform some modification with the belief of improvement. Considering an image or a video frame as a large Damaged Image Block (DIB) is placed in the CNN as shown in Fig.2. Each DIB cell can be further subdivided into different Image Blocks (IBs), each of which may or may not contain damaged pixels. Furthermore, each Image Block is subdivided into several

Pixel Blocks (PBs) which are elementary objects to be inpainted. The recursive algorithm is shown in Algorithm-1

The basic assumption is that color difference is a strong indication of the degree of details in an Image Block. The threshold value alpha (α) sets the criterion on whether a recursive call to the CNN based inpainting algorithm is required (the algorithm terminates when the DIB is small). In our implementation, the value of α is a percentage of color variance of an IB (i.e., the maximum var(IB) is 100). If the color variance of IB is greater than the threshold value α, the algorithm is called recursively to handle the next level of details. Otherwise, the algorithm further divides an IB into several pixel blocks in the CNN cell (i.e., PBs). Another criterion is the percentage of damaged pixels.

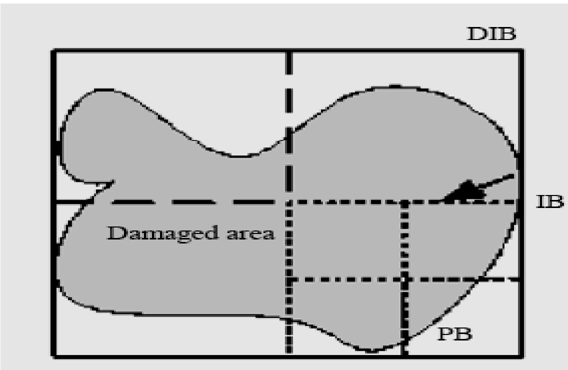

Fig-2 Damaged Image Block (DIB), Image Block(IB) and Pixel Block (PB) with CNN cell

We argue that if the percentage is too high, using surrounding color information to fix a pixel is less realistic as compared to using a global average color. In some severe cases, it is very difficult to use neighbouring cell colors. Note that, both thresholds are adjustable for the sake of analysis. The algorithm iterates through each of the PBs in an IB. If the percentage of damaged pixels in a PB is too high (i.e., greater than $\beta2$), the mean IB color is used [i.e., Mean Color (IB)]. One example is that the entire PB is damaged, so we must use the mean IB color. The function Damage Percent (PB) simply counts the number of damaged pixels in a PB. And, the function Fill (PB, C) takes a Pixel Block and a color C, and fills the Pixel Block with the color. Alternatively, if the percentage is still high (i.e., greater than $\beta1$), the mean PB color is used.

Algorithm:

```
Inpainting ( Damaged Image Block ( DIB ) )
{
Damaged Image Block = ε
    or
For All (IB) ε (DIB). ( var ( IB ) > α then Inpainting( IB )
    or
For all ( Pixel Block ) ε Image Block
 If ( ( Damaged Percent ( Pixel Block ) > β2
    Then Fill ( Pixel Block, Mean Color ( Image Block ) )
Else If ( β2 > = Damaged Percent ( Pixel Block ) > β1 ) then
    Fill ( Pixel Block, Mean Color ( Pixel Block ))
Else // Find Contour Matching for damaged Portion and check Same matching
within the image anywhere //
If ( Pattern Found ) then Fill ( Pixel Block damaged image with mean color ( Found
Pattern))
```

Algorithm-1 Proposed Algorithm for CNN based Inpainting Method.

Suppose a portion of an image or more number of cells is damaged and surrounding information not easy to get to inpaint or $\beta1$ and $\beta2$ which are too greater than threshold value, one can find the contour matching (chain code) values of the damaged cells and check with damaged pattern found in the same image. If the contour matching values of damaged image and the pattern found in the surrounding are same, the damaged portion is to be inpainted using mean color of the pixels of the pattern found in the surrounding anywhere else. Note that, the computation of mean colors does not take damaged pixels into account. If the percentage is low enough (i.e., less than $\beta1$), neighbour pixels and cells are used for interpolation. The function Interpolate (PB) implemented in our algorithm uses a bi-linear interpolation technique. We further take an optimization step to improve the algorithm. When the filling function is called, we add noise on the boundaries of Pixel Blocks. Our proposed CNN based image inpainting algorithm is called again to remove these bounding boxes. Thus, block effect is reduced. There are three thresholds in the above algorithm, α, $\beta1$ and $\beta2$ also used various combinations. The selection of $\beta2$ is aimed at testing the usage of mean color. Unless a pixel block is completely damaged, the mean color should be used. Thus, the selection of $\beta2$ should be

high. Since $\beta1 < \beta2$, we select the values of $\beta1$ accordingly. The threshold α is used to check the color variance. We try to cover a wide spectrum. The combinations of the above thresholds are all tested using more than 1000 pictures. The values of α, $\beta1$ and $\beta2$ have great impact on the outcome. In general, if α is less than 75, the average PSNR values of repaired pictures with respect to other parameters are stable. So we use α =80 in our implementation of an automatic inpainting tool. We chose $\beta2$=95 through our analysis. This means that unless the percentage of damaged pixels in a pixel block is higher than 95, the mean color of an outside big block should not be used. The value of $\beta1$ is critical. If $\beta1$ is less than 60, the result is not as good as expected. However, the PSNR values of the fixed pictures become stable when the value of $\beta1$ becomes 80, 85, or 90. Conclusively, we choose α =80, $\beta1$=85, and $\beta2$=95.

IV. EXPECTED RESULT

Proposed method will improve the visual quality of image and it will be able to restore the images with high noise ratio, because cellular neural network proved to be very useful regarding real-time image processing. Our objective is not to detect spikes or long vertical lines commonly reported in the literature. We aim to develop a technique to recover images with very high percentage of noise and achieve visually good output.

The key factor of proposed method is no. of damaged pixels, color variance and shape tracking. When dealing with a region or object, several compact representations is available that can facilitate manipulation of and measurements on the object. This paper extends the previous approaches with the use of shape tracking method that has been used by researchers in biological vision. Researchers in biological vision have long hypothesized that image contours (ordered sets of edge pixels, or contour points) are a compact yet descriptive representation of object shape. In computer vision, there has been substantial interest in extracting contours from images as well as using object models based on contours for object recognition, and 3D image interpretation.

V. CONCLUSION

After surveying different methods, it is observed that, Considerable computing power is necessary to solve the image processing task described by variational computing. The findings of our perceptual evaluation of inpainting techniques yielded valuable information related to the quality of the inpainting. Combining this information with low level and mid level image features we are currently involved in introducing an objective computational model for automatically predicting the perceptual image inpainting quality.

For the time being serial processing does not provide us with methods implementable in real-time. The cellular neural network proved to be very useful regarding real-time image processing. The reduction of computing time, due to parallel processing, can be obtained only if the processing algorithm can be implemented on CNN-UC.

Even if variational methods are used, the determination of templates ensuring the gray-scale image and the desired processing remains a difficult problem, since the fact that the actually existing CNN chip can use only linear templates having 3*3 dimension has to be taken into consideration. In some cases templates satisfy these conditions by using nonlinear templates. Effective CNN implementation is still possible in CNN algorithms.

Open Problems

In this paper, we have surveyed all the recent inpainting techniques, based on that we deduced a model for combining Cellular Neural Network Ideas with different images. The classical CNN approach for Digital Image Inpainting framework has proven to be very effective in designing and improving image inpainting. We have explained that the fundamental techniques for inpaintings such as Isophotes (line of equal gray value), PDE, Navier – Stokes equation, Total Variation (TV) methods, and Curvature Driven Diffusion (CCD) ideas in the Literature Survey. As a result, it increases the execution time due to performing computationally expensive operations again and again. Since the serial processing methods are not suitable for real-time operations of image, our proposed CNN technique is the best one for real time image inpainting due to its parallel computation. Finally we post two interesting open problems.

(i) **Video inpainting-** Video inpainting is a crucial and challenging area of image processing. It has profound applications in the movie industry, surveillance analysis, and dynamic vision analysis. The first open problem is: How to calculate and analyze the Peak Signal Noise Ratio (PSNR) values for video images with Cellular Neural Network as tool.

(ii) **Digital realization-** Throughout our problem formulation, we found Numerical PDE has been a core computational tool for all the geometric inpainting models. The second open problem concerns fast and efficient digital implementation of the associated PDE's, especially for the high order ones and also we can utilize Cellular Neural Network Techniques for efficient digital realization.

REFERENCES

[1] P.Elango and Dr. K.Murugesan " *Digital Image inpainting using cellular Neural Network*", International journal open problems in Computer Sciences and Mathematics(IJOPCM), Vol 2, No3, pp.439-450.
[2] P. Elango and Dr. K. Murugesan "*Image Restoration Using Cellular Neural Network With Contour Tracking Ideas*", International Journal of Computer Theory and Engineering, Vol. 2, No. 5, October, 2010 1793-8201

[3] L. O. Chua, and L. Yang, "*Cellular Neural Networks: Theory and Applications*", IEEE Trans. on Circuits and Systems, Vol.35, 1998, pp. 1257- 1290.
[4] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "*Simultaneous structure and texture image inpainting*", IEEE Transactions on Image Processing, vol. 12(8), 2003, pp.882-889.
[5] M. Bertalmio, G Sapiro, V. Caselles, and C. Ballester, "*Image Inpainting*". ACM SIGGRAPH Conference on Computer Graphics, SIGGRAPH 2000, pp.417-424.
[6] M. Oliveira, B. Bowen, R McKenna, and Y. S. Chang, "*Fast Digital Image Inpainting*". International Conference on Visualization, Imaging and Image Processing, 2001, pp. 261-266.
[7] L.O. Chua, and L. Yang, "*Cellular Neural Networks: Theory and Applications*", IEEE Trans. on Circuits and Systems, Vol.35, 1998, pp. 1257-1290.
[8] T. Roska, and L.O. Chua, "*The CNN universal machine an analogic array computer*", IEEE Trans. on Circuits and Systems, Vol. 40, 1993, pp. 163-173.
[9] L.C.Evans, "*Partial Differential Equations (PDE)*" , Graduate Studies in Mathematics. 19. Providence RI: American Mathematical Society (AMS),1998.
[10] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J.Verdera, "*Filling-in by joint interpolation of vector fields and gray levels*", IEEE Trans. Image Processing, vol. 10(1), 2001, pp. 1200-1211.
[11] T. Chan, and J. Shen, "*Mathematical Models for Local Non texture Inpaintings*", SIAM Journal of Applied Mathematics, vol. 62(3), 2001, pp. 1019-1043. [12] J. Shen, "*Inpainting and the fundamental problem of image processing*", SIAM News, vol. 36(5), 2003, pp. 110-121.
[13] V. Murugesh, and K. Murugesan, "*Comparison of Numerical Integration in Raster CNN Simulation*". LNCS, vol. 3285, 2004, pp. 115–122.
[14] T. Chan, and J. Shen, "*Non-texture inpainting by Curvature-Driven Diffusions (CCD)*", Journal of Visual Communication. Vol. 12, 2001, pp. 436–449.
[15] J. Kaipio, and E. Somersalo, "*Computational and Statistical Inverse Problems*", Springer, Berlin vol.160, 2005.
[16] Gacsadi, and P. Szolgay, "*Interpolation of 2D signals using CNN*", Proceedings of the European Conference on Circuit Theory and Design, (ECCTD'01), Espoo, 2001, pp. 349-352.
[17] Jian-Feng Cai, Raymond H. Chan, Zuowei Shen "*A framelet-based image inpainting algorithm*" Applied and Computational Harmonic Analysis, Volume 24, Issue 2, March 2008, pp. 131-149.
[18] Charles K. Chui "*An MRA approach to surface completion and image inpainting*" Applied and Computational Harmonic Analysis, Volume 26, Issue 2, March 2009, pp. 270-276.
[19] Lucia Maddalena,Alfredo Petrosino and Giuliano Laccetti "*A fusion-based approach to digital movie restoration*" Pattern Recognition, Volume 42, Issue 7, July 2009, pp.1485-1495.
[20] Dalong Li, , Steven Simske," *Shape retrieval with flat contour Segments"* Intelligent Enterprise Technologies Laboratory, HP Laboratories Palo Alto, HPL-2002-250, September 9th , 2002
[21] V. Casellas, J.M. Morel, G. Sapiro, and A. Tannenbaum, "*Introduction to the special issue on partial differential equations and geometry-driven diffusion in image processing and analysis*", IEEE Trans. On Image Processing, Vol. 7, 1998, pp. 269-273.